JEGYZŐKÖNYV

Adatkezelés XML környezetben Féléves feladat Vadásztársaság

Készítette: Jordán Kornél Neptunkód: DWWAP

Dátum: 2022.11.10.

A feladatban egy vadászvizsga lebonyolítását biztosító adatbázis modellt fogok bemutatni.

**Az egyedei**: Egyesület, Felügyelő, Vadász, Vizsga

# Egyesület egyed:

Az Egyesület egyed az adott vadászvizsga lebonyolítását biztosító egyesületnek az adatait

tartalmazza. Van benne: E\_ID, ami kulcs típusú, E\_Név, alapítás\_Éve, Telekhely.

# Felügyelő egyed:

A felügyelő egyed azokat a személyeket mutatja be, akik felügyelik az adott vizsgázót. Van benne: Személyi\_IG, ami kulcs típusú, Név ebben van Vezetéknév és Keresztnév,

Telefonszám, Email\_cím

# Vadász egyed:

A vadász egyedben az adott vizsgázó személyek adatai vannak benne. Van benne:

Személyi\_IG ami kulcs típusú, Név ebben van Vezetéknév és Keresztnév, Kor, Telefonszám, Email\_cím, Elért\_pontszám.

# Vizsga egyed:

A vizsga egyed az adott vizsga adatait tárolja. Van benne: Vizsga\_ID ami kulcs típusú, van Elmélet amiben van Írásbeli és Szóbeli rész. Van még Gyakorlat, Helyszín, amiben van

Irányító\_szám, Település, amiben van Utca és Házszám.

# Kapcsolatok:

**Vizsga – Egyesület:** 1:N kapcsolat, mivel egy vizsga csak egy egyesülethez tartozhat, de egy egyesületnek lehet több vizsgája is. (Egyesület-vizsga)

**Vadász – Egyesület:** 1:1 kapcsolat, mivel egy egyesületnek csak egy elnöke lehet (Elnök)

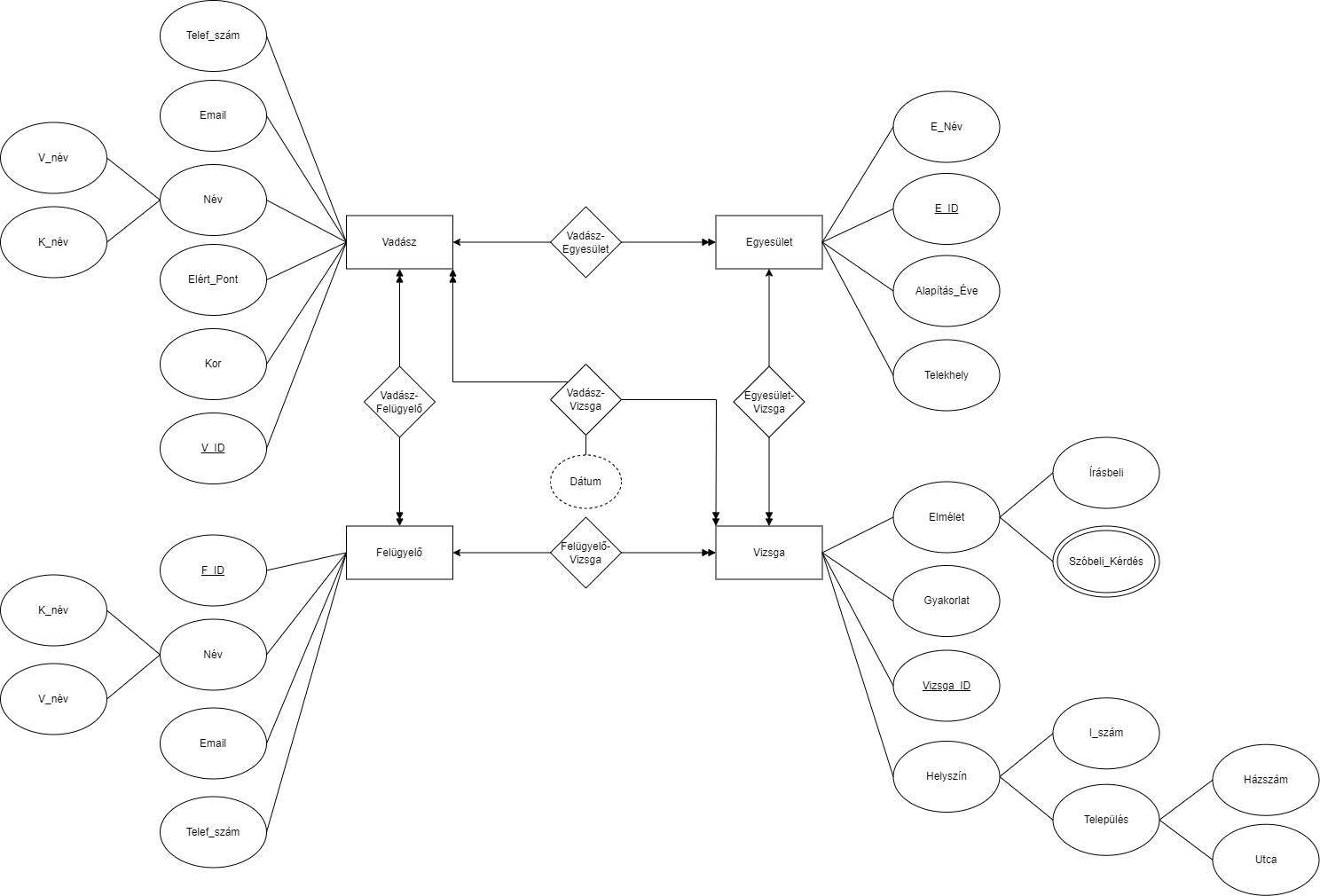
**Vadász – Felügyelő:** N:M kapcsolat, mivel egy vadásznak lehet több felügyelője is és egy felügyelő felügyelni tud több vadászt is. (Vadász-felügyelő)

**Vadász – Vizsga:** N:M kapcsolat, mivel több vadász jelentkezhet több vizsgára. (Vadász- vizsga)

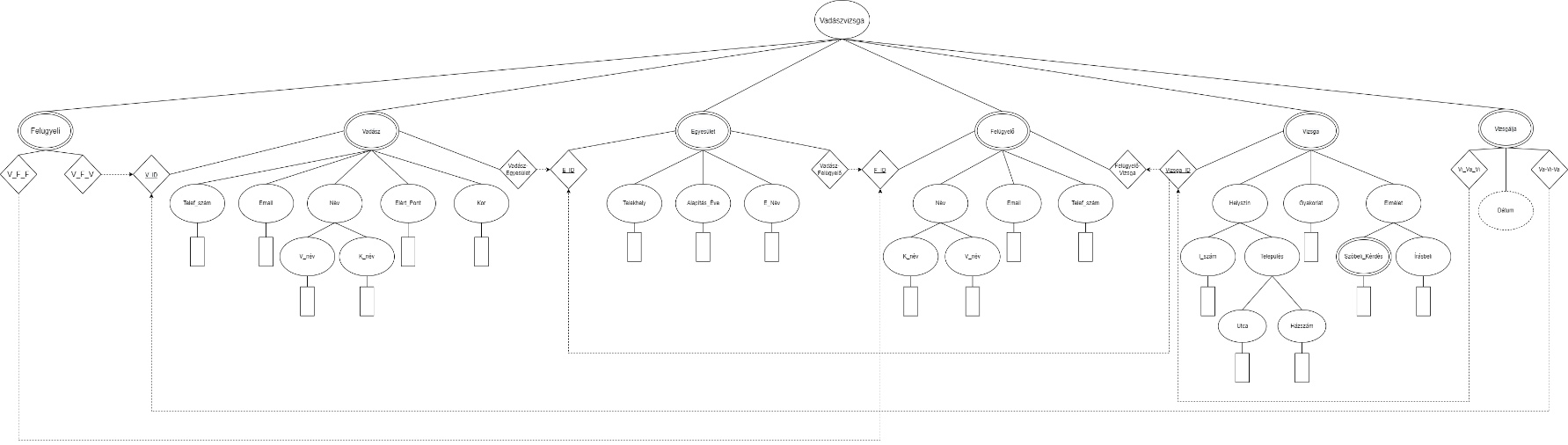
**Vizsga – Felügyelő:** 1:N kapcsolat, mivel egy vizsga az több felügyelőhöz tartozik, de egy felügyelő csak egy vizsgán tud egyszerre részt venni. (felügyelő-vizsga)

A Szóbeli\_Kérdés egy többértékű tulajdonság, mivel annak van Sorszám és Szöveg része is.

**1a) Az adatbázis ER modell: A modell egy vadásztársaságon belüli vizsgát mutat be**

1.

**1b) Az adatbázis konvertálása XDM modellre: Az 1a) feladatban megadott XDM modell konvertálása szabvány alapján**



**1c) Az XDM modell alapján XML dokumentum készítése: Az 1b) feladat ábrájának átírása XML dokumentumra**

<?xml version="1.0" encoding="UTF-8"?>

<vadaszvizsga xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="XMLSchemaDWWUAP.xsd">

<vadasz vadasz-egyesulet="ve1" V\_ID="1">

    <telef\_szam>06202222222</telef\_szam>

    <Email>valami@Email.com</Email>

    <V\_nev>Kiss</V\_nev>

    <K\_nev>Pista</K\_nev>

    <elert\_pont>5</elert\_pont>

    <kor>30</kor>

</vadasz>

<vadasz vadasz-egyesulet="ve2" V\_ID="2">

    <telef\_szam>06202222223</telef\_szam>

    <Email>valami2@Email.com</Email>

    <V\_nev>Nagy</V\_nev>

    <K\_nev>Pista</K\_nev>

    <elert\_pont>6</elert\_pont>

    <kor>40</kor>

</vadasz>

<vadasz vadasz-egyesulet="ve3" V\_ID="3">

    <telef\_szam>06202222224</telef\_szam>

    <Email>valam3@Email.com</Email>

        <V\_nev>Valamilyen</V\_nev>

        <K\_nev>Pista</K\_nev>

    <elert\_pont>7</elert\_pont>

    <kor>50</kor>

</vadasz>

<egyesulet E\_ID="1" vadasz-felugyelo="1">

    <telekhely>Miskolc</telekhely>

    <alapitas\_eve>1969</alapitas\_eve>

    <e\_nev>Kis Egyesület</e\_nev>

</egyesulet>

<egyesulet E\_ID="2" vadasz-felugyelo="2">

    <telekhely>Pest</telekhely>

    <alapitas\_eve>1980</alapitas\_eve>

    <e\_nev>Nagy Egyesület</e\_nev>

</egyesulet>

<egyesulet E\_ID="3" vadasz-felugyelo="3">

    <telekhely>Debrecen</telekhely>

    <alapitas\_eve>1990</alapitas\_eve>

    <e\_nev>Gyors Egyesület</e\_nev>

</egyesulet>

<felugyelo F\_ID="1" felugyelo-vizsga="1">

    <V\_nev>Kiss</V\_nev>

    <K\_nev>Lajos</K\_nev>

    <Email>lajos@gmail.com</Email>

    <telef\_szam>06203334444</telef\_szam>

</felugyelo>

<felugyelo F\_ID="2" felugyelo-vizsga="2">

    <V\_nev>Nagy</V\_nev>

    <K\_nev>Lajos</K\_nev>

    <Email>lajosanagy@gmail.com</Email>

    <telef\_szam>06203334456</telef\_szam>

</felugyelo>

<felugyelo F\_ID="3" felugyelo-vizsga="3">

    <V\_nev>Kiss</V\_nev>

    <K\_nev>Ferenc</K\_nev>

    <Email>ferenc@gmail.com</Email>

    <telef\_szam>06203338888</telef\_szam>

</felugyelo>

<vizsga vizsga\_ID="1">

    <I\_szam>3666</I\_szam>

    <utca>Petőfi</utca>

    <hazszam>1</hazszam>

    <gyakorlat>Kinti</gyakorlat>

    <szobeli\_kerdes>Valami1</szobeli\_kerdes>

    <irasbeli>Valami3</irasbeli>

</vizsga>

<vizsga vizsga\_ID="2">

    <I\_szam>3700</I\_szam>

    <utca>Nagy</utca>

    <hazszam>7</hazszam>

    <gyakorlat>Kinti</gyakorlat>

    <szobeli\_kerdes>Valami7</szobeli\_kerdes>

    <irasbeli>Valami4</irasbeli>

</vizsga>

<vizsga vizsga\_ID="3">

    <I\_szam>3800</I\_szam>

    <utca>Fekete</utca>

    <hazszam>30</hazszam>

    <gyakorlat>Kinti</gyakorlat>

    <szobeli\_kerdes>Valami3</szobeli\_kerdes>

    <irasbeli>Valami8</irasbeli>

</vizsga>

<vizsgalja Vi\_Va\_Vi="1" Va\_Vi\_Va="1">

<datum>2022.11.09.</datum>

</vizsgalja>

<vizsgalja Vi\_Va\_Vi="2" Va\_Vi\_Va="2">

    <datum>2022.11.16.</datum>

</vizsgalja>

<vizsgalja Vi\_Va\_Vi="3" Va\_Vi\_Va="3">

    <datum>2022.11.23.</datum>

</vizsgalja>

<felugyeli V\_F\_F="VFF1" V\_F\_V="VFV1"></felugyeli>

<felugyeli V\_F\_F="VFF2" V\_F\_V="VFV2"></felugyeli>

<felugyeli V\_F\_F="VFF3" V\_F\_V="VFV3"></felugyeli>

</vadaszvizsga>

**1d) Az XML dokumentum alapján XMLSchema készítése (saját típusok, ref, key, keyref, speciális**

**elemek): Az adatok kommentezve vannak a könnyebb olvashatóság miatt**

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<!-- Adatok típusai-->

<xs:element name="telef\_szam" type="xs:integer" />

<xs:element name="Email" type="xs:string" />

<xs:element name="V\_nev" type="xs:string" />

<xs:element name="K\_nev" type="xs:string" />

<xs:element name="elert\_pont" type="xs:integer" />

<xs:element name="kor" type="xs:integer" />

<xs:element name="telekhely" type="xs:string" />

<xs:element name="alapitas\_eve" type="xs:integer" />

<xs:element name="e\_nev" type="xs:string" />

<xs:element name="I\_szam" type="xs:integer" />

<xs:element name="utca" type="xs:string" />

<xs:element name="hazszam" type="xs:integer" />

<xs:element name="gyakorlat" type="xs:string" />

<xs:element name="szobeli\_kerdes" type="xs:string" />

<xs:element name="irasbeli" type="xs:string" />

<xs:element name="datum" type="xs:string" />

<xs:attribute name="V\_F\_F" type="xs:string" />

<xs:attribute name="V\_F\_V" type="xs:string" />

<xs:attribute name="V\_ID" type="xs:integer" />

<xs:attribute name="vadasz-egyesulet" type="xs:string" />

<xs:attribute name="E\_ID" type="xs:integer" />

<xs:attribute name="vadasz-felugyelo" type="xs:integer" />

<xs:attribute name="F\_ID" type="xs:integer" />

<xs:attribute name="felugyelo-vizsga" type="xs:integer" />

<xs:attribute name="vizsga\_ID" type="xs:integer" />

<xs:attribute name="Vi\_Va\_Vi" type="xs:integer" />

<xs:attribute name="Va\_Vi\_Va" type="xs:integer" />

<!-- Komplex típusok-->

    <xs:complexType name="vadaszTipus">

        <xs:sequence>

            <xs:element ref="telef\_szam"/>

            <xs:element ref="Email"/>

            <xs:element ref="V\_nev"/>

            <xs:element ref="K\_nev"/>

            <xs:element ref="elert\_pont"/>

            <xs:element ref="kor"/>

        </xs:sequence>

        <xs:attribute ref="V\_ID" use="required"/>

        <xs:attribute ref="vadasz-egyesulet" use="required"/>

    </xs:complexType>

    <xs:complexType name="egyesuletTipus">

        <xs:sequence>

            <xs:element ref="telekhely"/>

            <xs:element ref="alapitas\_eve"/>

            <xs:element ref="e\_nev"/>

        </xs:sequence>

        <xs:attribute ref="vadasz-felugyelo" use="required"/>

        <xs:attribute ref="E\_ID" use="required"/>

    </xs:complexType>

    <xs:complexType name="felugyeloTipus">

        <xs:sequence>

            <xs:element ref="V\_nev"/>

            <xs:element ref="K\_nev"/>

            <xs:element ref="Email"/>

            <xs:element ref="telef\_szam"/>

        </xs:sequence>

        <xs:attribute ref="felugyelo-vizsga" use="required"/>

        <xs:attribute ref="F\_ID" use="required"/>

    </xs:complexType>

    <xs:complexType name="vizsgaTipus">

        <xs:sequence>

            <xs:element ref="I\_szam"/>

            <xs:element ref="utca"/>

            <xs:element ref="hazszam"/>

            <xs:element ref="gyakorlat"/>

            <xs:element ref="szobeli\_kerdes"/>

            <xs:element ref="irasbeli"/>

        </xs:sequence>

        <xs:attribute ref="vizsga\_ID" use="required"/>

    </xs:complexType>

    <xs:complexType name="vizsgaljaTipus">

        <xs:sequence>

            <xs:element ref="datum"/>

        </xs:sequence>

        <xs:attribute ref="Vi\_Va\_Vi" use="required"/>

        <xs:attribute ref="Va\_Vi\_Va" use="required"/>

    </xs:complexType>

    <xs:complexType name="felugyeliTipus">

        <xs:attribute ref="V\_F\_F" use="required"/>

        <xs:attribute ref="V\_F\_V" use="required"/>

    </xs:complexType>

<xs:element name="vadaszvizsga">

    <xs:complexType>

        <xs:sequence>

            <xs:element name="vadasz" type="vadaszTipus" maxOccurs="unbounded" />

            <xs:element name="egyesulet" type="egyesuletTipus" maxOccurs="unbounded" />

            <xs:element name="felugyelo" type="felugyeloTipus" maxOccurs="unbounded" />

            <xs:element name="vizsga" type="vizsgaTipus" maxOccurs="unbounded" />

            <xs:element name="vizsgalja" type="vizsgaljaTipus" maxOccurs="unbounded" />

            <xs:element name="felugyeli" type="felugyeliTipus" maxOccurs="unbounded" />

        </xs:sequence>

    </xs:complexType>

<!-- Kulcsok -->

    <xs:key name="Vadasz\_kulcs">

        <xs:selector xpath="vadasz" />

        <xs:field xpath="@V\_ID" />

    </xs:key>

    <xs:key name="egyesulet\_kulcs">

        <xs:selector xpath="egyesulet" />

        <xs:field xpath="@E\_ID" />

    </xs:key>

    <xs:key name="felugyelo\_kulcs">

        <xs:selector xpath="felugyelo" />

        <xs:field xpath="@F\_ID" />

    </xs:key>

    <xs:key name="vizsga\_kulcs">

        <xs:selector xpath="vizsga" />

        <xs:field xpath="@vizsga\_ID" />

    </xs:key>

    <!-- Idegen kulcsok-->

    <xs:keyref refer="Vadasz\_kulcs" name="vadasz\_idegen\_kulcs">

        <xs:selector xpath="egyesulet" />

        <xs:field xpath="@vadasz-egyesulet" />

    </xs:keyref>

    <xs:keyref refer="egyesulet\_kulcs" name="vizsga\_idegen\_kulcs">

        <xs:selector xpath="felugyelo" />

        <xs:field xpath="@vadasz-felugyelo" />

    </xs:keyref>

    <xs:keyref refer="felugyelo\_kulcs" name="vizsga2\_idegen\_kulcs">

        <xs:selector xpath="felugyeli" />

        <xs:field xpath="@felugyelo-vizsga" />

    </xs:keyref>

    <xs:keyref refer="felugyelo\_kulcs" name="vizsga3\_idegen\_kulcs">

        <xs:selector xpath="vizsga" />

        <xs:field xpath="@Vi\_Va\_Vi" />

    </xs:keyref>

    <xs:keyref refer="felugyelo\_kulcs" name="vadasz2\_idegen\_kulcs">

        <xs:selector xpath="vadasz" />

        <xs:field xpath="@Va\_Vi\_Va" />

    </xs:keyref>

    <xs:keyref refer="felugyelo\_kulcs" name="felugyelo\_idegen\_kulcs">

        <xs:selector xpath="felugyelo" />

        <xs:field xpath="@V\_F\_F" />

    </xs:keyref>

    <xs:keyref refer="vizsga\_kulcs" name="vadasz3\_idegen\_kulcs">

        <xs:selector xpath="vadasz" />

        <xs:field xpath="@V\_F\_V" />

    </xs:keyref>

</xs:element>

</xs:schema>

**2. feladat**

**2a) adatolvasás: Egy olvasó létrehozása, bejövő adatok ”formájának” megadása, azért, hogy minden adatot megfelelően olvasson be.**

package hu.domparse.dwwuap;

import java.io.File;

import java.io.IOException;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import javax.xml.parsers.ParserConfigurationException;

import org.w3c.dom.Document;

import org.w3c.dom.Element;

import org.w3c.dom.Node;

import org.w3c.dom.NodeList;

import org.xml.sax.SAXException;

public class DOMReadDWWUAP {

    public static void main(String[] args) throws ParserConfigurationException, SAXException, IOException {

        //Forrás file

        File file = new File("XMLDWWUAP.xml");

        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();

        DocumentBuilder dBuilder = factory.newDocumentBuilder();

        Document doc = dBuilder.parse(file);

        doc.getDocumentElement().normalize();

        //Gyökérelem

        System.out.println("Root element: " + doc.getDocumentElement().getNodeName());

        //Gyerekelemek lementése

        NodeList nList = (NodeList) doc.getDocumentElement();

        for (int i = 0; i < nList.getLength(); i++) {

            Node node = nList.item(i);

            //Ha egyesület

            if(node.getNodeName() == "egyesulet"){

                if(!node.getNodeName().equals("#text")) {

                    System.out.println("\n");

                    System.out.println("Current element: " + node.getNodeName());

                }

                //Egyesület adatainak kiírása

                if(node.getNodeType()==Node.ELEMENT\_NODE) {

                    Element elem = (Element) node;

                    String e\_id = elem.getAttribute("E\_ID");

                    Node egyesulet\_node = elem.getElementsByTagName("telekhely").item(0);

                    String telekhely\_name = egyesulet\_node.getTextContent();

                    Node egyesulet\_node2 = elem.getElementsByTagName("alapitas\_eve").item(0);

                    String alapitas\_eve\_name = egyesulet\_node2.getTextContent();

                    Node egyesulet\_node3 = elem.getElementsByTagName("e\_nev").item(0);

                    String e\_nev\_name = egyesulet\_node3.getTextContent();

                    System.out.printf("Egyesulet id: %s%n", e\_id);

                    System.out.printf("Telekhely neve: %s%n", telekhely\_name);

                    System.out.printf("Alapitas eve: %s%n", alapitas\_eve\_name);

                    System.out.printf("Egyesulet neve: %s%n", e\_nev\_name);

                }

            }

        }

    }

}

**2b) adatmódosítás: Alap file beolvasása, mindig az adott kommentek alapján történő műveletek csinálása, az adattagokkal.**

package hu.domparse.dwwuap;

import java.io.File;

import java.io.IOException;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import javax.xml.parsers.ParserConfigurationException;

import javax.xml.transform.Transformer;

import javax.xml.transform.TransformerFactory;

import javax.xml.transform.dom.DOMSource;

import javax.xml.transform.stream.StreamResult;

import org.w3c.dom.Document;

import org.w3c.dom.Element;

import org.w3c.dom.NamedNodeMap;

import org.w3c.dom.Node;

import org.w3c.dom.NodeList;

import org.xml.sax.SAXException;

public class DOMModifyDWWUAP {

    public static void main(String[] args) throws ParserConfigurationException, SAXException, IOException {

        try {

            //Forrás file

            File inputFile = new File("XML2DWWUAP.xml");

            DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();

            DocumentBuilder docBuilder = docFactory.newDocumentBuilder();

            Document doc = docBuilder.parse(inputFile);

            //ELső és harmadik egyesület mentése

            Node vadasz1 = doc.getElementsByTagName("vadasz").item(0);

            Node vadasz3 = doc.getElementsByTagName("vadasz").item(2);

            //Gyökérelem

            Node vadaszvizsga = doc.getFirstChild();

            //Harmadik egyesület ID váltás

            NamedNodeMap attr = vadasz3.getAttributes();

            Node nodeAttr = attr.getNamedItem("V\_ID");

            nodeAttr.setTextContent("20");

            //Második egyesület irányítószámának megváltoztatása

            NodeList list = vadasz1.getChildNodes();

            for (int i = 0; i < list.getLength(); i++) {

                Node node = list.item(i);

                if (node.getNodeType() == Node.ELEMENT\_NODE) {

                    Element elem = (Element) node;

                    if ("telef\_szam".equals(elem.getNodeName())) {

                        elem.setTextContent("06206964200");

                    }

                }

            }

            //Harmadik egyesületnek az Email megváltoztatása

            NodeList list1 = vadasz3.getChildNodes();

            for (int i = 0; i < list1.getLength(); i++) {

                Node node1 = list1.item(i);

                if (node1.getNodeType() == Node.ELEMENT\_NODE) {

                    Element elem1 = (Element) node1;

                    if ("Email".equals(elem1.getNodeName())) {

                        elem1.setTextContent("valamiModify@Email.com");

                    }

                }

            }

            //Egyesületek törlése

            NodeList childNodes = vadaszvizsga.getChildNodes();

            for(int i = 0; i < childNodes.getLength(); i++) {

                Node node = childNodes.item(i);

                if("egyesulet".equals(node.getNodeName()))

                    vadaszvizsga.removeChild(node);

            }

            //Konzolra kírás

            TransformerFactory transformerFactory = TransformerFactory.newInstance();

            Transformer transformer = transformerFactory.newTransformer();

            DOMSource source = new DOMSource(doc);

            System.out.println("New File");

            StreamResult consoleResult = new StreamResult(System.out);

            transformer.transform(source, consoleResult);

            }catch(Exception e) {

                e.printStackTrace();

            }

        }

}

**2c) adatlekérdezés: Alap beolvasó létrehozása, adatok belementése, végig megyünk az elemein kihagyva a kort.**

package hu.domparse.dwwuap;

import java.io.File;

import java.io.IOException;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import javax.xml.parsers.ParserConfigurationException;

import org.w3c.dom.Document;

import org.w3c.dom.Element;

import org.w3c.dom.Node;

import org.w3c.dom.NodeList;

import org.xml.sax.SAXException;

public class DOMQueryDWWUAP {

    public static void main(String[] args) throws ParserConfigurationException, SAXException, IOException {

        //Forrás file

        File file = new File("XMLDWWUAP.xml");

        DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();

        DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();

        Document doc = dBuilder.parse(file);

        doc.getDocumentElement().normalize();

        //Gyökér elem

        System.out.print("Root element: ");

        System.out.println(doc.getDocumentElement().getNodeName());

        //Vadaszvizsga betöltése

        NodeList nList = doc.getElementsByTagName("vadaszvizsga");

        System.out.println("-----------------");

        //Végigfut a vadasznak a gyerek elemein kihagyva a kort

        for (int i = 0; i < nList.getLength(); i++) {

            Node node = nList.item(i);

            System.out.println("\nCurrent Element : "+node.getNodeName());

            if (node.getNodeType() == Node.ELEMENT\_NODE) {

                Element elem = (Element) node;

                System.out.println("ID:"+elem.getAttribute("V\_ID"));

                NodeList nList2 = elem.getChildNodes();

                for (int j = 0; j < nList2.getLength(); j++) {

                    Node node2 = nList2.item(j);

                    if (node2.getNodeType() == Node.ELEMENT\_NODE) {

                        Element elem2 = (Element) node2;

                        if(!node2.getNodeName().equals("kor")) {

                            System.out.println(node2.getNodeName()+" : "+node2.getTextContent());

                        }

                        NodeList nList3 = elem2.getChildNodes();

                        for (int k = 0; k < nList3.getLength(); k++) {

                            Node node3 = nList3.item(k);

                            if(node3.getNodeType()==Node.ELEMENT\_NODE) {

                                System.out.println("vadaszvizsga :  "+node3.getNodeName()+" : "+node3.getTextContent());

                            }

                        }

                    }

                }

            }

        }

    }

}